

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Currently Amended) A transfer control protocol (TCP) transmission system, comprising:

a transmit request handler that receives request events from a requester including a notification of new transmission data posted, records event information into a connection context and, based on the notification of new transmission data posted, either schedules a connection in a ready queue or places the connection in a pending queue; and

a transmitter that operates in parallel with the transmit request handler, wherein the transmitter dequeues connections from the ready queue and prepares packets for transmission based on information recorded in the connection context.

2. (Original) The TCP transmission system of claim 1, wherein the ready queue comprises a linked list of connections.

3. (Original) The TCP transmission system of claim 1, wherein each connection comprises a connection context data structure.

4. (Canceled)

5. (Original) The TCP transmission system of claim 1, wherein the request events include a request to send an acknowledgement.
6. (Original) The TCP transmission system of claim 1, wherein the request events include a request for a window update.
7. (Original) The TCP transmission system of claim 1, wherein the request events include handling of an incoming acknowledgement.
8. (Currently Amended) The TCP transmission system of claim 1, further comprising a queue manager, wherein the queue manager includes ~~a system~~ means for receiving a timer expiration.
9. (Currently Amended) The TCP transmission system of claim 1, wherein the transmitter includes:
 - ~~a system~~ means for deciding what type of segment should be transmitted;
 - ~~a system~~ means for building a transmit command and requesting segment data;
 - and
 - ~~a system~~ means for building header information for the segment being transmitted.
10. (Currently Amended) A method for transmitting packets in a transfer control

protocol (TCP) environment, comprising:

submitting receiving a request event to at a transmit request handler, the request event including a notification of new transmission data posted;

processing the request event in the transmit request handler to, based on the notification of new transmission data posted, either schedule a connection in a ready queue or place the connection in a pending queue;

providing a transmitter that operates in parallel with the transmit request handler;
and

utilizing the transmitter to dequeue connections from the ready queue and prepare packets for transmission.

11. (Currently Amended) The method of claim 10, wherein the request event is further selected from the group consisting of: ~~a notification of new transmission data posted~~, a request to send an acknowledgement, a request for a window update, and a request to handle an incoming acknowledgement.

12. (Currently Amended) The method of claim 10, further comprising ~~the further step of~~ moving connections from the pending queue to the ready queue.

13. (Currently Amended) The method of claim 10, wherein the ~~step of~~ dequeuing connections from the ready queue and prepare packets for transmission, ~~includes the steps of~~:

- handling any timeouts of a retransmit timer;
- deciding on a type of segment to transmit;
- building a transmit command and requesting segment data;
- building headers;
- recording information on a previously transmitted segment; and
- starting a retransmit timer if data was transmitted.

14. (Original) The method of claim 13, wherein each connection includes data describing the request event in a connection context.

15. (Currently Amended) A system for transmitting packets in a transfer control protocol (TCP) environment, comprising:

- a connection context for storing event information;
- a transmit request handler that receives request events from a requester including a notification of new transmission data posted, records the event information into the connection context and, based on the notification of new transmission data posted, either schedules a connection in a ready queue or places the connection in a pending queue;
- a transmitter that operates in parallel with the transmit request handler, wherein the transmitter dequeues connections from the ready queue and prepares packets for transmission based on event information stored in the connection context; and
- a queue manager for moving connections from the pending queue to the ready

queue.

16. (Original) The system of claim 15, wherein the ready queue comprises a linked list of connections.

17. (Original) The system of claim 15, wherein each connection comprises a connection context data structure.

18. (Original) The system of claim 15, wherein the request event is selected from the group consisting of: a notification of new transmission data posted, a request to send an acknowledgement, a request for a window update, and a request to handle an incoming acknowledgement.

19. (Original) The system of claim 15, wherein the transmitter includes:

means for deciding what type of segment should be transmitted;

means for building a transmit command and requesting segment data; and

means for building header information for the segment being transmitted.

20. (Original) The system of claim 15, wherein the queue manager receives any timer expirations.